

README

DISCLAIMER:

*This font is still work in progress so please do not hesitate to email me regarding any issues or missing features at bee.trice.Olive@gmail.com * if you encounter them. I am also still a novice when it comes to ASL so some of my decisions regarding what features to include for the first few versions may not make a lot of sense. Your input will, I hope, lead to a better and more useable v2.0 in the future.*

Thank you,

-Bee 🐝🐝.

** That Olive is zero-l-i-v-e, I only mention it because I have lost a few emails due to a misreading before, and I don't want to seem like I'm just ignoring your emails.*

Welcome to ASLWrite Font v1.5!

CURRENT FEATURES

- 50 digits, with mirrored, rotated, raised, and lowered variants
- 64 movement marks with mirrored and rotated variants, and 8 strays
- Overlapping Characters
- Non Manual Marks

BEFORE USE

Please make sure the word processor you are using has support for ligatures. The end of the document has a compiled list of programs that support or should support the font, along with pros and cons for each one. Once you have your software of choice, please disable both spell checking and automatic capitalization, use a larger font size and maybe adjust line spacing, this one depends on your program.



This is your new keyboard! The shifted keys are represented in the lower half of the key, and the lowercase is set in the top half of the key. You're going to end up referring to the keyboard layout chart a lot, so if you're just starting out, printing out the image to keep next to you as reference might not be a bad idea. There are also a few companies that sell custom prints for key caps, and I'm working to make a template to send them for personal use, but if anyone wants the file when its done you could order a custom keyboard specifically for use with this font.

INSTALLATION

Download the .OTF, open, and click install. It will look a little goofy, but the installer is attempting to write a sentence in English.

HOW TO USE

This font relies heavily on the ligature system, and while I tried to make it intuitive, I've only ever made English fonts before. As a result, there are definitely some idiosyncrasies in this system. Also be aware, you will be making heavy use of your space bar. The characters all, for the most part, have the same hit box, but because they're different sizes of characters, it means they overlap. This is important for the system to function correctly, but it also makes spacing things a little more difficult when typing.

SPACING

If everything transfers correctly, the Space Bar, lowercase "a" and lowercase "z" will be used to space characters apart from eachother. Lowercase "a" has the same width as any other character, but is empty. Lowercase "z" has 1/2 the width as "a". Space bar has three times the width as "a". Together, they should allow rather fine control of character spacing.

		
space	a	z
 .		
space space	aa	zz
 .		
space space space	aaa	zzzzzz

BODY MARKS

The body marks are on your lowercase number bar, from the back tick to the equals sign.

`	1	2	3	33	4	5	6	7	8	9	99	0	00	-	=
(	┐	┌	>	»	}	}	⤿	⤿	)	(	┌	┌	┌	}	>

CONTACT MARKS

⤿.	⤿.	⤿.	⤿.	⤿.
TPP	TP	T	TO	TOO
⤿..	⤿..	⤿..	⤿..	⤿..
TTPP	TTP	TT	TTO	TTOO
⤿...	⤿...	⤿...	⤿...	⤿...
TTTPP	TTTP	TTT	TTTO	TTTTOO
⤿...				
TTTT				

				
TTXP	TTX	TTXO		
				
TTTTP	TTTX	TTTXO		
				
TTTTT				

NON MANUAL MARKS

There are 14 currently supported faces, alongside the shoulder shift mark.

Smile	Frown	Flat Lips	Pressed Lips	Tongue Out	Lips Out	Pursed Lips	Open Mouth
1@	1@s	1~	1~~	1!	1!!	1!!s	1^
							
Bared Teeth	"Cha"	Round Mouth	Puffed Cheeks	Puffed & Round	Pulled to Side	Shoulder Shift	
1^#	1^#!	1%	1\$	1%\$	1%#	9GF	
							

DIACRITICS AND PUNCTUATION

At the moment, the eyes don't merge with faces, that may be coming, but I am not sure how to make the spacing work yet.

⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
&_	*_	*s_	(_	)_	&+	*+	*s+	(+	)+	"
⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
'''	''''	'	"	'''	?	??	???	/	//	]
⌘	⌘	⌘	⌘	⌘	⌘					
]]	[	[[	;	q	w					

DIGIBET CHARACTERS

Digits are constructed from 19 keys, and are written in a specific order each time:

[BASE][FINGERS][MIRROR][ROTATE]

BASE

These characters constitute the smallest visual component of a digit that cannot be broken down further.

⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
r	t	y	u	i	f	g	h	j	v	b	n	m

FINGERS

These keys inform the system how it is supposed to form a readable key. The "g" here is both a base and a modifying key. Digits are constructed in the shown left to right order.

						
d	o	k	,	e	c	g

COMPLETE DIGITS

				
r	t	te	tk	y
				
yd	yc	uk	ud	ug
				
udk	ugk	i	ie	f
				
fk	f,	g,	h,	hk
				
hd	hdk	jo	jok	jo,
				
joe	j,g	j,	jk	je

				
jc	jcc	vk	v,	vc
				
vck	vcg	vckg	b	be
				
bk	bo	no	nok	no,
				
nk	n,	m	mk	mkk

MIRROR

When a character would be written as the opposite hand, the "s" key can be used to mirror almost every digit. Digits like  and  would not have mirrored variants because they would still be the same shape. Some examples of mirrored digits:

			
jok	nok	uk	vc
			
joks	noks	uks	vcs

ROTATE

There are currently eight directions of rotation. Most characters can be considered to be pointing up, but a couple are facing to the side. The keys "WERFCXZA" are used to designate the angle the hand should face.

Up Facing Digits							
g,	g,R	g,F	g,C	g,X	g,Z	g,A	g,W
Left Facing Digits							
iE	iR	iF	iC	iX	iZ	i	iW
Down Facing Digits							
mkE	mkR	mkF	mkC	mk	mkZ	mkA	iW

Digits Facing Left By Default:

vk	i	t	te	b	bo	bk	hk

Down Facing By Default Digits:

mk	mkk

MOVEMENT MARKS

Movement marks are constructed in a similar manner as digits.

[BASE][PLANE][FOLLOW ALONG][DOT][MIRROR][ROTATE]

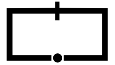
Your bases are bound to G, H, HH, J, V, B, N, M, while the Plane Change mark is bound to S, and the Follow Along rotation mark is bound to D. The following movement marks can be mirrored and rotated in all eight directions. All rotatable movement marks are considered Up Facing.

ᵢ	ᵢ	ᵢ	ᵢ	ᵢ	ᵢ	ᵢ	ᵢ
G	GT	GS	GST	GD	GDT	GSD	GSdT
↺	↺	↺	↺	↺	↺	↺	↺
H	HT	HS	HST	HD	HDT	HSD	HSdT
↻	↻	↻	↻	↻	↻	↻	↻
HH	HHT	HHS	HHST	HHD	HHDT	HHS D	HHSDT
ᵐ	ᵐ	ᵐ	ᵐ	ᵐ	ᵐ	ᵐ	ᵐ
J	JT	JS	JST	JD	JDT	JSD	JSdT
ᶜ	ᶜ	ᶜ	ᶜ	ᶜ	ᶜ	ᶜ	ᶜ
V	VT	VS	VST	VD	VDT	VSD	VSDT
⌢	⌢	⌢	⌢	⌢	⌢	⌢	⌢
B	BT	BS	BST	BD	BSD	BSD	BSDT

							
N	NT	NS	NST	ND	NDT	NSD	NSDT
							
M	MT	MS	MST	MD	MDT	MSD	MSDT

CONSTRUCTED MARKS

Like the digits, some movement marks are constructed by combining others together. These marks do not yet have rotational variants, so what's here is all there is at the moment.

							
TB	BB	BBB	BBS	BBBS	BSB	BSBT	MSM

MORPH MARKS

I think these might count as movement marks, but because they serve slightly different purposes and are typed in a different manner they're being separated out here. These are probably some of the most annoying characters to get used to, they're designed so that they overlap with themselves, and can be drawn really long underneath other characters, the problem being you basically need to write these characters between the others and if you need to adjust your spacing after the first pass it gets hectic fast.

>	>>	>>>
<	<<	<<<
>P	>>P	>>>P
<P	<<P	<<<P

Lengthening Example
jok > >h,>a>
Two Handed Example


joks >Pjoka>P >P>h,sa>Pa>aa>>h,>a>

I am truly sorry for how messy this gets, but its the only way I could think of that actually worked.

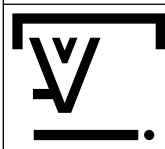
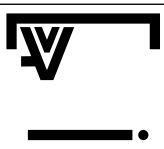
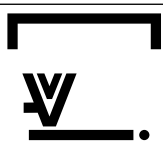
Contact Morph

<S aa>

This is not a ligature, and the spacing and length can be edited to your desired parameters.

STACKING

All digits , after being mirrored and rotated to the desired orientation, may be raised and lowered with the same keys that move the contact marks, capital O and P.

Standard	Raised	Lowered	Example
			
jok	jokO	jokP	ugFO TTO ugsAP

the digits should still all fit within the bounds of the shoulder and morph marks, even rotated. The characters can't be stacked directly on top of each other, but this is close as we can get.

So far, only three movement marks can be manipulated in this way, more may follow.

		
GO	GXP	NFO

On Text Editors and Word Processors

In trying to get this font working and out there, its become apparent that the ligature system, while perfect for this use, is not something that a lot of word processors support. Some systems that support ligatures include:

WORD PROCESSORS

Zoho Writer- Primarily a browser word processor, but there is a desktop version of the application. Locally installed fonts can be added to the program, but are not uploaded automatically. To add a font to Zoho, while in a document, click on the settings gear in the top right corner, go to Fonts, then Custom Fonts. You need to type the name of the font that the file itself uses, which is "ASLWrite", Add, save, and you're good to go. Zoho Writer does not seem to have the ability to export to PDF with fonts intact however.

Microsoft Word - Has ligature support, but must be turned on in the settings. Costs \$160, or a monthly fee for the office suite, or you could pirate it. Just be careful. PDF export with fonts intact should be possible.

Libre Office - Ligatures are on by default, but the program *can* be unstable with custom fonts and large images. Save often just in case. Can export PDFs with fonts intact.

TEXT EDITORS

Strangely, you can actually use a number of text editor programs, which are distinct from word processors in that Text Editors are primarily used for programming. But its that use case that actually tends to allow for ligatures in the first place, as

programmers will often use fonts with specially designed ligatures to make some of their code more legible. I haven't done a whole lot of looking into which of these will support ligatures, but Visual Studio, and VS Code do. Not actually sure what the difference between the programs is but I'm sure someone does.

IMAGE EDITORS

Image and vector manipulation software like Adobe Illustrator and Photoshop should both support font ligatures, mainly for their uses in designing logos and signage, not as useful for day to day applications, but great for an aslwrite infographic or similar!

ASLWriteFont was made using Glyphr Studio, and contains an estimated 3,260 characters as of v1.7. If you've got experience making fonts, and you want to see the Glyphr file to make a fork or something, let me know and I'll send it on over